

# Level Set Segmentation From Multiple Non-Uniform Volume Datasets

Ken Museth<sup>†\*</sup>

David E. Breen<sup>†</sup>

Leonid Zhukov<sup>‡</sup>

Ross T. Whitaker<sup>‡</sup>

<sup>†</sup>Computer Science Department  
California Institute of Technology

<sup>‡</sup>School of Computing  
University of Utah

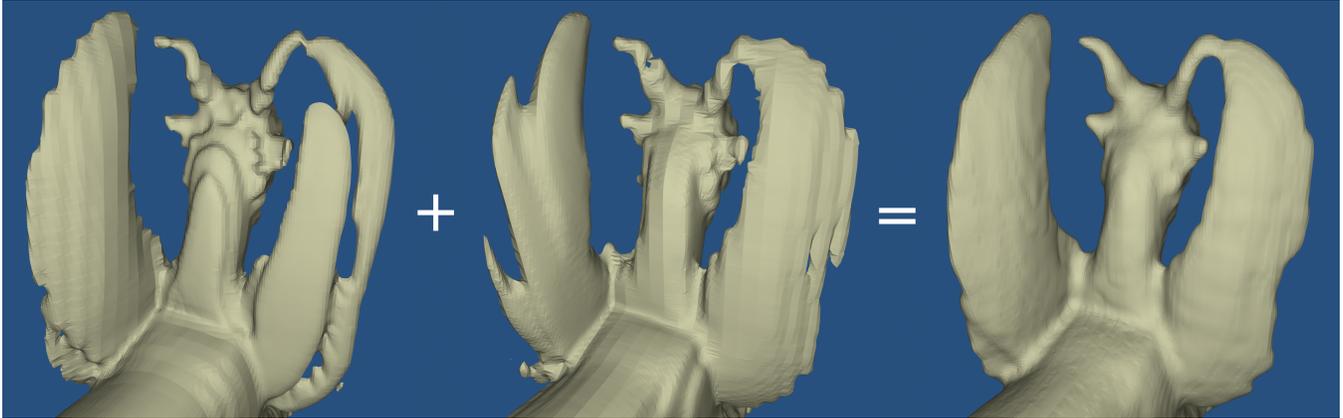


Figure 1: 3D level set model of a griffin derived from two non-uniform laser scan reconstruction volume datasets. The two input models with resolution  $294 \times 312 \times 24$  and  $294 \times 52 \times 144$  show severe aliasing effects from insufficient sampling, especially in the wings. Our method merges information from both of the datasets to produce a high resolution  $294 \times 312 \times 144$  level set model. All models throughout this paper are flat-shaded to highlight details.

## ABSTRACT

Typically 3-D MR and CT scans have a relatively high resolution in the scanning  $X - Y$  plane, but much lower resolution in the axial  $Z$  direction. This non-uniform sampling of an object can miss small or thin structures. One way to address this problem is to scan the same object from multiple directions. In this paper we describe a method for deforming a level set model using velocity information derived from multiple volume datasets with non-uniform resolution in order to produce a single high-resolution 3D model. The method locally approximates the values of the multiple datasets by fitting a distance-weighted polynomial using moving least-squares. The proposed method has several advantageous properties: its computational cost is proportional to the object surface area, it is stable with respect to noise, imperfect registrations and abrupt changes in the data, it provides gain-correction, and it employs a distance-based weighting to ensure that the contributions from each scan are properly merged into the final result. We have demonstrated the effectiveness of our approach on four multi-scan datasets, a griffin laser scan reconstruction, a CT scan of a teapot and MR scans of a mouse embryo and a zucchini.

**CR Categories:** I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Surface and object representations;

**Keywords:** Segmentation, visualization, level set models, 3D reconstruction.

\*kmu@caltech.edu

## 1 INTRODUCTION

Many of today’s volumetric datasets are generated by medical MR, CT and other scanners. A typical 3-D scan has a relatively high resolution in the scanning  $X - Y$  plane, but much lower resolution in the axial  $Z$  direction. The difference in resolution between the in-plane and out-of-plane samplings can easily range between a factor of 5 to 10, see figure 1. This occurs both because of physical constraints on the thickness of the tissue to be excited during scanning (MR), total tissue irradiation (CT), and scanning time restrictions. Even when time is not an issue, most scanners are by design incapable of sampling with high resolution in the out-of-plane direction, producing anisotropic “brick-like” voxels.

The non-uniform sampling of an object or a patient can create certain problems. The inadequate resolution in the  $Z$  direction implies that small or thin structures will not be properly sampled, making it difficult to capture them during surface reconstruction and object segmentation. One way to address this problem is to scan the same object from multiple directions, with the hope that the small structures will be adequately sampled in one of the scans. Generating several scans of the same object then raises the question of how to properly combine the information contained in these multiple datasets. Simply merging the individual scans does not necessarily assemble enough samples to produce a high resolution volumetric model. Normally a technique for filling in between samples is needed.

We have previously developed a framework for extracting 3D models from volume datasets [23] based on level set methods [13]. In this paper we describe a method for deforming a level set model using velocity information derived from multiple volume datasets with non-uniform resolution in order to produce a single high-resolution 3D model. The method locally approximates the values of the multiple datasets by fitting a distance-weighted polynomial

using moving least-squares (MLS) [19, 8]. Directional 3D edge information that may be used during a level set segmentation process is readily derived from MLS.

The proposed method has several advantageous properties. Instead of merging all of the input volumes by global resampling (interpolation), we locally approximate the derivatives of the intensity values by MLS. This local versus global approach is feasible because the segmentation process is implemented with a deformable level set model that requires only edge information in a narrow band around the surface. Consequently the MLS calculation is only performed in a small region of the volume, rather than throughout the whole volume, making the computational cost proportional to the object surface area [25]. As opposed to many interpolation schemes the MLS method is stable with respect to noise and imperfect registrations [5]. Our implementation also allows for small intensity attenuation artifacts between the multiple scans thereby providing gain-correction. The distance-based weighting employed in our method ensures that the contributions from each scan is properly merged into the final result. If a slice of data from one scan is closer to a point of interest on the model, the information from this scan will contribute more heavily to determining the location of the point.

To the best of our knowledge there is no previous work on creating deformable models directly from multiple volume datasets. While there has been previous work on 3D level set segmentation and reconstruction [24, 10, 9, 20, 26], it has not been based on multiple volume datasets. However, 3D models have been generated from multiple range maps [22, 4, 25, 16], but the 2D nature of these approaches is significantly different from the 3D problem being addressed in this paper. The most relevant related projects involve merging multiple volumes to produce a single high-resolution volume dataset [21, 6], and extracting edge information from a single non-uniform volume [2]. Our work does not attempt to produce a high-resolution merging of the input data. Instead, our contribution stands apart from previous work because it deforms a model based on local edge information derived from multiple non-uniform volume datasets.

We have demonstrated the effectiveness of our approach on four multi-scan datasets. The first three examples are derived from single high resolution volume datasets that have been sub-sampled in the  $X$ ,  $Y$  and  $Z$  directions respectively. Since the non-uniform scans are extracted from a single dataset they are therefore perfectly aligned. The first example is a volumetric laser scan reconstruction of a griffin model. The second example is a high resolution MR scan of a 12-day-old mouse embryo, which has already had its outer skin isolated with a previous segmentation process. The third example is a preprocessed high resolution CT scan of a teapot, that also only contains an outer surface. The final example consists of multiple MR scans of a zucchini that have been imperfectly aligned by hand. The first three examples show that our method is able to perform level set segmentation from multiple non-uniform scans of an object, picking up and merging features only found in one of the scans. The final example demonstrates that our method generates satisfactory results, even when there are misalignments in the registration.

The remainder of the paper has the following structure. In Section 2 we outline the details of our method, and in section 3 we present the results obtained with this method. We close with conclusions and an appendix describing the moving least-squares method.

## 2 METHOD DESCRIPTION

We formulate our approach to 3D reconstruction of geometric models from multiple non-uniform volumetric datasets within a level-set segmentation framework [23]. The level set models utilized

within this framework are deformable implicit surfaces whose deformation is controlled by a speed function in the level set partial differential equation (PDE). The speed function describes the velocity at each point on the evolving surface in the direction of the local surface normal. All of the information needed to deform a surface is encapsulated in the speed function, providing a simple, unified computational framework. In this section we briefly describe our level set segmentation framework, review the fundamental level set PDE, and define speed functions that allow us to solve the multiple-data segmentation problem. The key to constructing suitable speed terms is 3D directional edge information derived from the multiple datasets. This problem is solved using a moving least-squares scheme that extracts edge information by locally fitting sampling points to high-order polynomials. This section concludes by outlining the overall algorithm of the method.

### 2.1 Level Set Segmentation Framework

Level set segmentation relies on a surface-fitting strategy that creates a new volume from the input data by solving a partial differential equation (PDE) with user-defined feature-extracting terms. Because the deformable models move using gradient descent, they seek *local solutions*, and therefore the results are strongly dependent on the starting position of the surface. Thus, level set deformations alone are not sufficient, they must be combined with powerful initialization techniques in order to produce successful segmentations. Our level set segmentation framework consists of a set of suitable pre-processing techniques for initialization, which are then followed by the selection and tuning of different feature-extracting terms in the level set algorithm, as seen in Figure 2 [23]. Once these terms are defined the level set deformation proceeds to produce the final result.

Each stage in this two-step process is equally important for generating a correct segmentation. A user must “mix-and-match” these operations in order to produce the desired result. The operators available for creating the initial model include high and low thresholding, flood-filling, as well as CSG and morphological (opening and closing) operators. These operators provide a rough initial estimate of the desired model. The level set surface deformation process then moves the model toward specific features in the data while balancing this movement with a regularizing smoothing term, in order to prevent the surface from fitting too closely to noise-corrupted data. The smoothing term utilizes a local mean curvature measure in order to remove regions of high curvature. The inclusion of such a regularization term is also a well known technique in most implicit snake algorithms. Concurrently, the level set model may be attracted to “Canny” edges [3], iso-surfaces and regions of maximum gradient magnitude in the input data.

### 2.2 The Level Set Method

The Level Set Method [13] is a mathematical tool for modeling surface deformations. A deformable (*i.e.* time-dependent) surface,  $\mathcal{S}(t)$ , is implicitly represented as an iso-surface of a time-varying scalar function  $\phi(\mathbf{x}, t)$  embedded in 3D,<sup>1</sup> *i.e.*

$$\mathcal{S}(t) = \{\mathbf{x}(t) \mid \phi(\mathbf{x}(t), t) = k\}, \quad (1)$$

where  $k \in \mathbb{R}$  is the iso-value,  $t \in \mathbb{R}^+$  is time, and  $\mathbf{x}(t) \in \mathbb{R}^3$  is a point in space on the iso-surface. It might seem inefficient to implicitly represent a surface with a 3D scalar function; however the higher dimensionality of the representation provides one

<sup>1</sup>Our work uses the dynamic level set equation, which is more flexible than the corresponding stationary equation,  $\phi(\mathbf{x}) = k(t)$ , see [18] for more details.

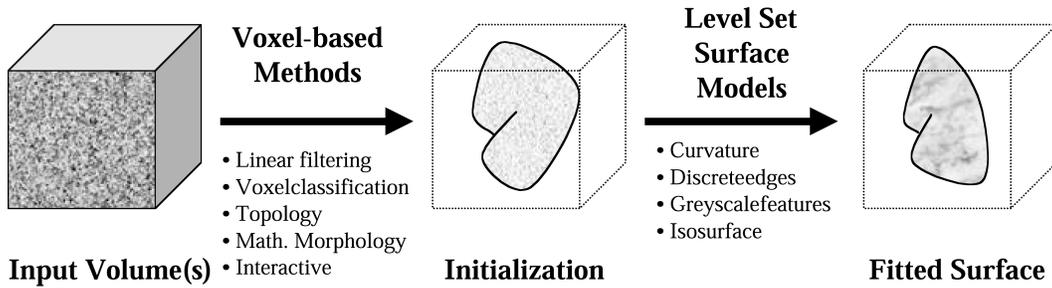


Figure 2: Level set segmentation stages – initialization and surface deformation.

of the major advantages of the LS method: the flexible handling of changes in the topological genus of the deformable surface. This implies that LS surfaces can easily represent complicated surface shapes that can form holes, split to form multiple objects, or merge with other objects to form a single structure. This is an important property when segmenting complex models with an unknown topological genus.

The fundamental level set equation of motion for  $\phi(\mathbf{x}(t), t)$  is derived by differentiating Eq. (1) with respect to time  $t$ , and applying the chain rule giving:

$$\frac{\partial \phi}{\partial t} = -\nabla \phi \cdot \frac{d\mathbf{x}}{dt} = \|\nabla \phi\| \mathcal{F}(\mathbf{x}, \mathbf{n}, \phi) \quad (2a)$$

$$\mathcal{F}(\mathbf{x}, \mathbf{n}, \phi) \equiv \mathbf{n} \cdot \frac{d\mathbf{x}}{dt}, \quad (2b)$$

where  $d\mathbf{x}/dt$  and  $\mathbf{n} \equiv -\nabla \phi / \|\nabla \phi\|$  are the velocity and normal vectors at  $\mathbf{x}$  on the surface. We assume a positive-inside/negative-outside sign convention for  $\phi(\mathbf{x}, t)$ , *i.e.*  $\mathbf{n}$  points outward. Eq. (2b) introduces the speed function  $\mathcal{F}$ , which is a user-defined scalar function that can depend on any number of variables including  $\mathbf{x}$ ,  $\mathbf{n}$ ,  $\phi$  and its derivatives evaluated at  $\mathbf{x}$ , as well as a variety of input data.  $\mathcal{F}()$  is a *signed* scalar function that defines the motion (*i.e.* speed) of the level set surface in the direction of the local normal  $\mathbf{n}$  at  $\mathbf{x}$ .

A number of numerical techniques [13, 1] make the initial value problem of Eq. (2) computationally feasible. A complete discussion of the details of the level set method is beyond the scope of this paper. We instead refer the interested reader to [18, 12, 17]. However, we will briefly mention two of the most important techniques: the first is the so called “up-wind scheme” which addresses the problem of overshooting when trying to integrate Eq. (2) in time by finite differences. Specifically the upwind scheme is used to compute first order partial derivatives by a single-sided finite difference which is up-wind with respect to the motion of the level set surface. The second important technique is related to the fact that one is typically only interested in a single solution to Eq. (2), say the  $k = 0$  level set. This implies that the evaluation of  $\phi$  is important only in the vicinity of a particular level set. This forms the basis for “narrow-band” schemes [1, 25, 14] that solve Eq. (2) in a narrow band of voxels near the surface. The “up-wind scheme” makes the level set method numerically robust, and the “narrow-band scheme” makes its computational complexity proportional to the level set’s surface area rather than the size of the volume in which it is embedded.

### 2.3 Level Set Speed Function for Segmentation

Many different speed functions have been proposed over the years for segmentation of a single volume dataset [24, 10, 9, 20]. Typically such speed functions consist of a (3D) image-based feature

attraction term and a smoothing term which serves as a regularization term that lowers the curvature and suppresses noise in the input data. From computer vision it is well known that features, *i.e.* significant changes in the intensity function, are conveniently described by an edge-detector [7]. There exists a very large body of work devoted to the problem of designing optimal edge detectors for 2D images [11, 3], most of which are readily generalized to 3D. For the work presented in this paper we found it convenient to use speed functions with the 3D directional edge term

$$\mathcal{F}_{edge}(\mathbf{x}, \mathbf{n}, \phi) = \alpha \mathbf{n} \cdot \nabla \|\nabla V_g\| \quad (3)$$

where  $\alpha$  is a scaling factor for the *image-based* feature attraction term  $\nabla \|\nabla V_g\|$ .  $V_g$  symbolizes some global uniform merging of the multiple non-uniform input volumes. This feature term is effectively a 3D directional edge-detector of  $V_g$ . However there are two problems associated with using this speed function exclusively. The first is related to the fact that we cannot expect to compute reliable 3D directional edge information in all regions of space simply because of the nature of the non-uniform multiple volumes that serves as input for our segmentation process. In other words  $V_g$  cannot be interpolated reliably in regions of space where there are no nearby sampling points. Hence the level set surface will not experience any image-based forces in these regions. In other words the surface fitting is an ill-posed problem in regions of space with no image-based information. The solution is to use a regularization term that imposes constraints on the mean curvature of the deforming level-set surface. We include the following smoothing term in the speed function in order to smooth the regions where no edge information exists as well as suppress noise in the remaining regions thereby preventing excessive aliasing;

$$\mathcal{F}_{smooth}(\mathbf{x}, \mathbf{n}, \phi) = \beta \nabla \cdot [\nabla \phi / \|\nabla \phi\|] \quad (4)$$

where  $\beta$  is a scaling factor for the mean curvature,  $\nabla \cdot [\nabla \phi / \|\nabla \phi\|]$ , on the level set surface defined from  $\phi$ .

However, one problem remains. Normally the feature attraction term,  $\nabla \|\nabla V_g\|$ , creates only a narrow range of influence. In other words, this feature attraction term will only reliably move the portion of the level set surface that is in close proximity to the actual edges in  $V_g$ . Thus, a good initialization of the level set surface is needed before solving Eq. (3). A reasonable initialization of the level set surface may be obtained by computing the CSG union of the multiple input volumes, which are first tri-linearly resampled to give a uniform sampling. However, if the input volumes are strongly non-uniform their union produces a poor initial model. This occurs when the input volumes are severely undersampled in one or more directions, as seen in Figure 3. Consequently we attract the CSG union surface to the distance transform of the Canny edges [3] computed from  $V_g$ . Canny edges are non-directional edges defined from the zero-crossing of the second derivative of the image



Figure 3: The union of the first three teapot datasets from Figure 6.

in the direction of the local normal. In 3D this is

$$\frac{\partial^2}{\partial \mathbf{n}_g^2} V_g = 0 \quad (5)$$

where  $\mathbf{n}_g \equiv \nabla V_g / \|\nabla V_g\|$  is the local normal vector of  $V_g$ . Using the expressions  $\partial / \partial \mathbf{n}_g = \mathbf{n}_g \cdot \nabla$  we can rewrite Eq. (5) as

$$\frac{\partial^2}{\partial \mathbf{n}_g^2} V_g = \mathbf{n}_g \cdot \nabla [\mathbf{n}_g \cdot \nabla V_g] = \mathbf{n}_g \cdot \nabla \|\nabla V_g\|. \quad (6)$$

This expression highlights the relationship between the Canny edge detector and the 3D directional edge detector defined in Eq. (3). The initialization procedure is then completed by pulling the CSG union of the multiple input volumes to the distance transform of the zero-crossing scalar Canny edge detector defined in Eq. (6).

The next section focuses on the methods needed to reliably compute the vectors  $\mathbf{n}_g$  and  $\nabla \|\nabla V_g\|$ . In preparation, the latter may be explicitly expressed in terms of the derivatives of the merged volume  $V_g$

$$\nabla \|\nabla V_g\| = \frac{\nabla V_g \hat{H} V_g}{\|\nabla V_g\|} \quad (7)$$

where we have defined the gradient vector and the Hessian matrix,

$$\hat{\nabla} V_g = \left( \frac{\partial V_g}{\partial x}, \frac{\partial V_g}{\partial y}, \frac{\partial V_g}{\partial z} \right) \quad (8a)$$

$$\hat{H} V_g = \begin{pmatrix} \frac{\partial^2 V_g}{\partial x^2} & \frac{\partial^2 V_g}{\partial y \partial x} & \frac{\partial^2 V_g}{\partial z \partial x} \\ \frac{\partial^2 V_g}{\partial x \partial y} & \frac{\partial^2 V_g}{\partial y^2} & \frac{\partial^2 V_g}{\partial z \partial y} \\ \frac{\partial^2 V_g}{\partial x \partial z} & \frac{\partial^2 V_g}{\partial y \partial z} & \frac{\partial^2 V_g}{\partial z^2} \end{pmatrix}. \quad (8b)$$

Thus, in closing we note that the level set propagation needed for segmentation only needs information about the first and second order partial derivatives of the input volumes, not the interpolated intensity values themselves.

## 2.4 Computing Partial Derivatives

As outlined above the speed function  $\mathcal{F}$  in the level-set equation, Eq. (2), is based on edge information derived from the input volumes. This requires estimating first and second order partial derivatives from the multiple non-uniform input volumes. We do this by means of moving least-squares (MLS), which is an effective and well established numerical technique for computing derivatives of functions whose values are known only on irregularly spaced points [19, 8, 5].

Let us assume we are given the input volumes  $\hat{V}_d$ ,  $d = 1, 2, \dots, D$  which are volumetric samplings of an object on the non-uniform grids  $\{\hat{\mathbf{x}}_d\}$ . We shall also assume that the local coordinate frames of  $\{\hat{\mathbf{x}}_d\}$  are scaled, rotated and translated with respect to each other. Hence, we define a world coordinate frame (typically one of the local frames) in which we solve the level set equation. Now, let us define the world sampling points  $\{\mathbf{x}_d\}$  as

$$\mathbf{x}_d \equiv \mathbf{T}^{(d)}[\hat{\mathbf{x}}_d] \quad (9)$$

where  $\mathbf{T}^{(d)}$  is the coordinate transformation from a local frame  $d$  to the world frame. Next we locally approximate the intensity values from the input volumes  $\hat{V}_d$  with a 3D polynomial expansion. Thus, we define the  $N$ -order polynomials

$$V_N^{(d)}(\mathbf{x}) = C_{000}^{(d)} + \sum_{i+j+k=1}^N C_{ijk}^{(0)} x^i y^j z^k, \quad d = 1, 2, \dots, D \quad (10)$$

where the  $C$  coefficients are unknown. Note that these local approximations to the intensity values share coefficients  $C_{ijk}^{(0)}$  of order higher than zero, *i.e.* all of the functions  $V_N^{(d)}$ ,  $d = 1, 2, \dots, D$  have the *same edges*. The fact that the zero-order term in Eq. (10) is input volume dependent means we allow for local constant offsets between the input volumes  $\hat{V}_d$ . This effectively provides built-in gain-correction in the scheme, since it can handle small intensity attenuation artifacts between the multiple scans. The details of deriving a set of linear equations for the coefficients  $C$  by means of the moving least-squares method is described in the Appendix. The resulting system of linear equations can be solved using standard techniques from numerical analysis. Summarizing the results from the Appendix, Eq. (18a) and Eq. (18b) can be conveniently expressed as

$$\sum_{\mathbf{q}} \mathbf{A}_{\mathbf{p}, \mathbf{q}} \mathbf{c}_{\mathbf{q}} = \mathbf{b}_{\mathbf{p}} \quad (11)$$

where  $\mathbf{A}$  is a diagonal matrix, and  $\mathbf{b}$ ,  $\mathbf{c}$  are vectors. In this equation we have also introduced the compact index notations  $\mathbf{p} \equiv (i, j, k, r)$  and  $\mathbf{q} \equiv (l, m, n, s)$  defined as

$$\mathbf{p} \in \{i, j, k, r \in \mathcal{N}^+ \mid i = j = k = 0, 1 \leq r \leq D\} \\ \cup \{i, j, k, r \in \mathcal{N}^+ \mid 1 \leq i + j + k \leq N, r = 0\} \quad (12a)$$

$$\mathbf{q} \in \{l, m, n, s \in \mathcal{N}^+ \mid l = m = n = 0, 1 \leq s \leq D\} \\ \cup \{l, m, n, s \in \mathcal{N}^+ \mid 1 \leq l + m + n \leq N, s = 0\}. \quad (12b)$$

The diagonal matrix  $\mathbf{A}$ , and the vectors  $\mathbf{b}$ ,  $\mathbf{c}$  in Eq. (11) are defined as

$$\mathbf{A}_{\mathbf{p}, \mathbf{q}} \equiv \sum_d (\delta_{r,d} + \delta_{r,0}) (\delta_{s,d} + \delta_{s,0}) \sum_{\mathbf{x}_d} w_d(\mathbf{x}_d - \mathbf{x}_0) \\ \times (\mathbf{x}_d - \mathbf{x}_0)^i (\mathbf{y}_d - \mathbf{y}_0)^j (\mathbf{z}_d - \mathbf{z}_0)^k \quad (13a)$$

$$\times (\mathbf{x}_d - \mathbf{x}_0)^l (\mathbf{y}_d - \mathbf{y}_0)^m (\mathbf{z}_d - \mathbf{z}_0)^n \\ \mathbf{b}_{\mathbf{p}} \equiv \sum_d (\delta_{r,d} + \delta_{r,0}) w_d(\mathbf{x}_d - \mathbf{x}_0) \hat{V}_d(\mathbf{x}_d) \\ \times (\mathbf{x}_d - \mathbf{x}_0)^i (\mathbf{y}_d - \mathbf{y}_0)^j (\mathbf{z}_d - \mathbf{z}_0)^k \quad (13b)$$

$$\mathbf{c}_{\mathbf{p}} \equiv C_{ijk}^{(r)}. \quad (13c)$$

Next the matrix equation  $\mathbf{A}\mathbf{c} = \mathbf{b}$  must be solved for the vector  $\mathbf{c}$  of dimension  $\binom{N+3}{3} + D - 1$ , where  $N$  is the order of the expansion in Eq. (10) and  $D$  is the number of non-uniform input volumes. As is well known for many moving least-square problems it is possible for the condition number of the matrix  $\mathbf{A}$  to become very large.

Any matrix is singular if its condition number is infinite and can be defined as ill-conditioned if the reciprocal of its condition number approaches the computer’s floating-point precision. This can occur if the problem is over-determined (number of sampling points,  $\mathbf{x}_d$  greater than number of coefficients  $C$ ) and under-determined (ambiguous combinations of the coefficients  $C$  work equally well or equally bad). To avoid such numerical problems, a singular value decomposition (SVD) linear equation solver is recommended for use in combination with the moving least-squares method. The SVD solver identifies equations in the matrix  $\mathbf{A}$  that are, within a specified tolerance, redundant (*i.e.* linear combinations of the remaining equations) and eliminates them thereby improving the condition number of the matrix. We refer the reader to reference [15] for a helpful discussion of SVD pertinent to linear least-squares problems.

Once we have the expansion coefficients  $\mathbf{c}$  we can readily express the Hessian matrix and the gradient vector of the combined input volumes as

$$\nabla V = (C_{100}^{(0)}, C_{010}^{(0)}, C_{001}^{(0)}) \quad (14a)$$

$$\mathbf{H}V = \begin{pmatrix} 2C_{200}^{(0)} & C_{110}^{(0)} & C_{101}^{(0)} \\ C_{110}^{(0)} & 2C_{020}^{(0)} & C_{011}^{(0)} \\ C_{101}^{(0)} & C_{011}^{(0)} & 2C_{002}^{(0)} \end{pmatrix} \quad (14b)$$

evaluated at the moving expansion point  $\mathbf{x}_0$ . This in turn is used in Eq. (7) to compute the edge information needed to drive the level set surface.

## 2.5 Algorithm Overview

The level set segmentation algorithm used in this paper is outlined below. Algorithm 2.1 describes the main steps of our approach. The initialization routine, Algorithm 2.2, is called for all of the multiple non-uniform input volumes,  $V_d$ . Each non-uniform input dataset is uniformly resampled using tri-linear interpolation. Edge information and the union,  $V_0$ , of the  $V_d$ ’s is then computed. Algorithm 2.2 calculates Canny and 3D directional edge information using moving least-squares in a narrow band in each of the resampled input volumes,  $V_d$ , and buffers this in  $V_{edge}$  and  $\mathbf{V}_{edge}$ . Next Algorithm 2.1 computes the distance transform of the zero-crossings of the Canny edges and takes the gradient of this scalar volume to produce a vector field  $\mathbf{V}_F$ , which pulls the initial level set model to the Canny edges. Finally the level set model is attracted to the 3D directional edges of the multiple input volumes,  $\mathbf{V}_{edge}$ , and a Marching Cubes mesh is extracted for visualization. The level set solver, described in Algorithm 2.3, solves Eq. (2) using the “upwind scheme” (not explicitly defined) and the sparse-field narrow-band method of [25], with  $V_0$  as the initialization and  $\mathbf{V}_F$  as the force field in the speed function.

### Algorithm 2.1: MAIN( $V_1, \dots, V_D$ )

**comment:**  $V_1, \dots, V_D$  are non-uniform samplings of object  $V$

**global**  $V_{edge}, \mathbf{V}_{edge}$

```

 $V_0 \leftarrow$  uniform sampling of empty space
for  $d \leftarrow 1$  to  $D$ 
  do  $V_0 \leftarrow V_0 \cup$  INITIALIZATION( $V_d$ )
   $\mathbf{V}_F \leftarrow \nabla$ [distance transform[zero-crossing[ $V_{edge}$ ]]]
   $V_0 \leftarrow$  SOLVELEVELSETEQ( $V_0, \mathbf{V}_F, \alpha, 0$ )
   $V_0 \leftarrow$  SOLVELEVELSETEQ( $V_0, \mathbf{V}_{edge}, \alpha, \beta$ )
return (Marching Cubes mesh of  $V_0$ )

```

### Algorithm 2.2: INITIALIZATION( $V_d$ )

**comment:** Pre-processing to produce good LS initialization

```

 $V_d \leftarrow$  Uniform tri-linear resampling of  $V_d$ 
 $\Gamma_d \leftarrow$  Set of voxels in narrow band of iso-surface of  $V_d$ 
do
  for each  $\mathbf{x}_0 \in \Gamma_d$ 
    do
      Solve moving least-squares problem at  $\mathbf{x}_0$ 
       $V_{edge}(\mathbf{x}_0) \leftarrow$  scalar Canny edge, cf. Eq. (6)
       $\mathbf{V}_{edge}(\mathbf{x}_0) \leftarrow$  3D directional edge, cf. Eq. (7)
return ( $V_d$ )

```

### Algorithm 2.3: SOLVELEVELSETEQ( $V_0, \mathbf{V}_F, \alpha, \beta$ )

**comment:** Solve Eq. (2) with initial condition  $\phi(t=0) = V_0$

```

 $\phi \leftarrow V_0$ 
repeat
   $\Gamma \leftarrow$  Set of voxels in narrow band of iso-surface of  $\phi$ 
   $\Delta t \leftarrow \gamma / \sup_{\mathbf{x} \in \Gamma} \|\mathbf{V}_F(\mathbf{x})\|$ ,  $\gamma \leq 1$ 
  do
    for each  $\mathbf{x} \in \Gamma$ 
       $\mathbf{n} \leftarrow$  upwind scheme[ $-\nabla\phi(\mathbf{x}) / \|\nabla\phi(\mathbf{x})\|$ ]
      do
         $\dot{\phi}(\mathbf{x}) \leftarrow \|\nabla\phi(\mathbf{x})\|(\alpha\mathbf{V}_F(\mathbf{x}) \cdot \mathbf{n} + \beta\nabla \cdot \mathbf{n})$ 
         $\phi(\mathbf{x}) \leftarrow \phi(\mathbf{x}) + \dot{\phi}(\mathbf{x})\Delta t$ 
    until  $\sup_{\mathbf{x} \in \Gamma} \|\dot{\phi}(\mathbf{x})\| \leq \epsilon$ 
return ( $\phi$ )

```

## 3 RESULTS

We have applied our segmentation method to several multi-scan non-uniform datasets to produce high resolution level set models. The parameters used for these segmentations are listed in Table 1.  $\alpha$  and  $\beta$  are weights that the user adjusts to balance attraction to edges with curvature-based smoothing during the level set deformation process.

Table 1: Maximum in-plane to out-of-plane sampling ratios of non-uniform input datasets, and parameters for the two level set speed terms defined in Eq. (3) and Eq. (4).

Model	Origin	Ratio	$\alpha$	$\beta$
Griffin	Laser scan	10:1	1.0	0.5
Mouse	MR scan	10:1	1.0	0.5
Teapot	CT scan	9:1	0.5	1.0
Zucchini	MR scan	10:1	1.0	0.5

### 3.1 Griffin Dataset

The griffin dataset was created with a volumetric laser scan reconstruction algorithm [4]. This algorithm creates a high resolution volumetric representation of an object by merging multiple depth maps produced via a laser scan. The original griffin dataset has a resolution of  $312 \times 294 \times 144$ . We have extracted three non-uniform datasets from this high resolution representation by copying every sixth plane of data in the  $X$  and  $Y$  directions and every tenth plane of data in the  $Z$  direction. The three derived non-uniform griffin datasets have the following resolution:  $52 \times 294 \times 144$ ,  $312 \times 30 \times 144$  and  $312 \times 294 \times 24$ . Iso-surfaces have been extracted from these datasets, appropriately scaled in the low resolution direction, and are presented in the first three images in Figure 4. Each low resolution scan inadequately captures some important geometric feature of the griffin. In the first scan the wing on the right contains numerous holes. In the second scan the horns on the head are not properly represented, and in the third image the wing on the left contains significant notches. Additionally, all three scans are severely

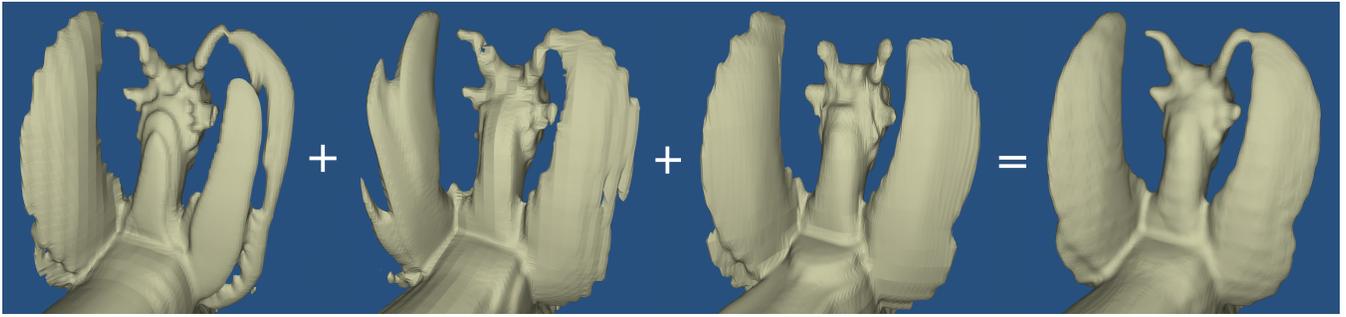


Figure 4: Three non-uniform samplings of a high resolution laser scan reconstruction of a griffin figurine, followed by a level set model derived from the first three scans. Each input model is missing a particular feature - first: holes in right wing, second: jagged edges of both wings, third: right horn not connected to wing (as it should be). The level set reconstruction contains all of these missing features.

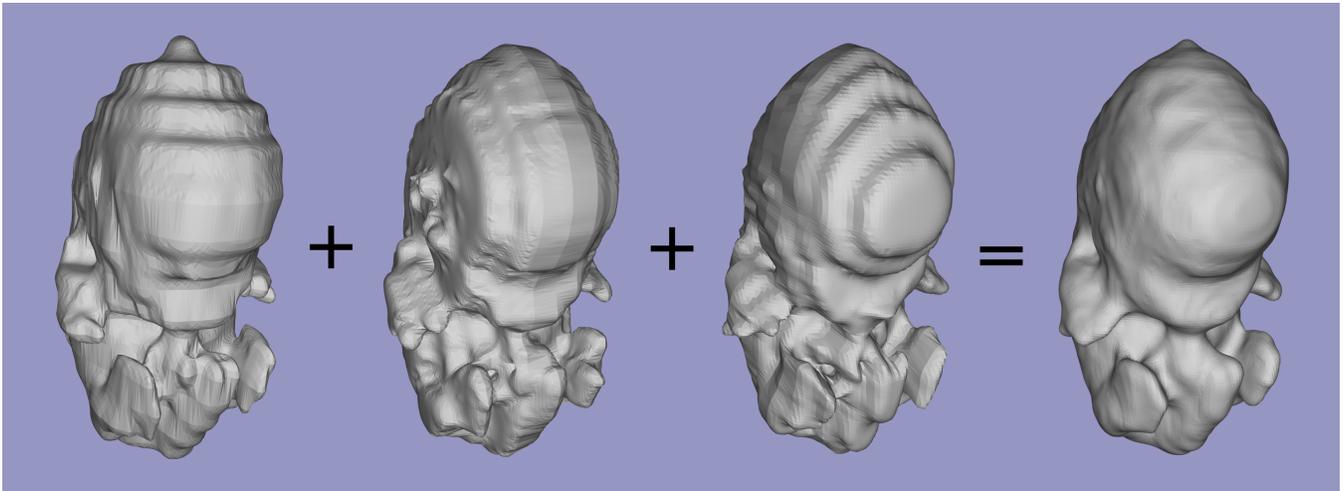


Figure 5: Three non-uniform samplings of a high resolution MR scan of a mouse embryo, followed by a level set model derived from the first three scans.

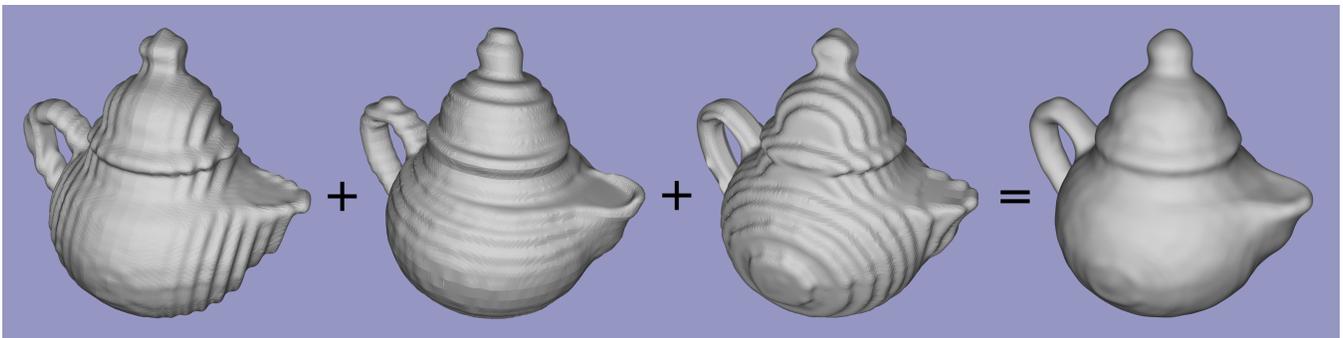


Figure 6: Three non-uniform samplings of a high resolution CT scan of a teapot, followed by a level set model derived from the first three scans.

aliased. We have performed two reconstructions from the under-sampled non-uniform scans. In Figure 1 a reconstruction produced from just the first two scans is presented. The final image in Figure 4 presents the results of applying our segmentation method to all three low resolution scans. The method produces high resolution ( $312 \times 294 \times 144$ ) level set models that contain all of the features mentioned above and do not exhibit the aliasing seen in the low resolution scans. Adding the third scan provides more information around the edges of the wings. It should also be noted that the wing on the right is connected to the right horn in the initial high resolution dataset.

### 3.2 Mouse Embryo Dataset

The first three scans in Figure 5 are derived from a high resolution MR scan of a mouse embryo. They are subsampled versions of a  $256 \times 128 \times 128$  volume dataset, and have the following resolutions:  $26 \times 128 \times 128$ ,  $256 \times 16 \times 128$  and  $256 \times 128 \times 13$ . The last image in Figure 5 presents the result produced by our multi-scan segmentation method. The information in the first three scans has been successfully used to create a level set model of the embryo with a resolution of  $256 \times 128 \times 130$ . The finer features of the mouse embryo, namely its hands and feet, have been reconstructed.

### 3.3 Teapot Dataset

The first three scans in Figure 6 are derived from a CT scan of a teapot. They are subsampled versions of a  $244 \times 218 \times 188$  volume dataset, and have the following resolutions:  $28 \times 218 \times 188$ ,  $244 \times 25 \times 188$  and  $244 \times 218 \times 21$ . The last image in Figure 6 presents the result produced by our multi-scan segmentation method. The information in the first three scans has been successfully used to create a level set model of the original teapot with a resolution of  $244 \times 218 \times 189$ . The finer features of the teapot, namely the handle and the spout, have been reconstructed.

### 3.4 Zucchini Dataset

The zucchini dataset consists of three individual MRI scans of an actual zucchini. The separate scans have been registered manually and are presented on the left side of Figure 7, each with a different color to demonstrate their imperfect alignment. The resolutions of the individual scans are  $28 \times 218 \times 188$ ,  $244 \times 25 \times 188$  and  $244 \times 218 \times 21$ . This image highlights the rough alignment of the scans. The right side of Figure 7 presents the result of our level set segmentation. It demonstrates that our approach is able to extract a reasonable model from multiple datasets that are imperfectly aligned.

## 4 CONCLUSIONS

In this paper we have proposed a method that uses multiple volume datasets with non-uniform resolution acquired in different local coordinate frames, but with known relative transformations, to deform a level set model on a uniform grid. As described in section 2.4, the contribution from each of the datasets to the velocity of the evolving level set model is weighted according to the quality (resolution) of the dataset near the propagating front. We obtain this result by employing a moving least-squares (MLS) method. Our method only performs the MLS calculation in the neighborhood of the propagating front and thus has  $O(N^2)$  computational complexity. Additionally, it is stable with respect to noise, imperfect registration and abrupt changes in the data, it provides gain-correction, and employs a distance-based weighting to ensure that the contributions from each scan are properly merged into the final result. We

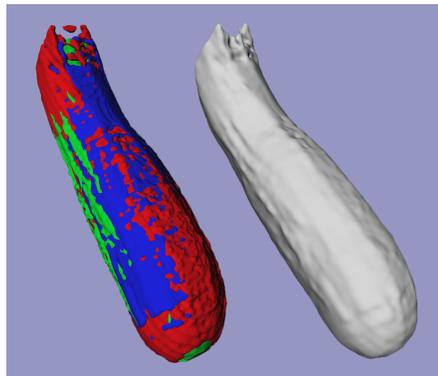


Figure 7: Three low resolution MR scans of a zucchini that have been individually colored and overlaid to demonstrate their imperfect alignment. The level set model on the right is derived from the three low resolution scans.

have demonstrated the effectiveness of our approach on four multi-scan datasets, a griffin laser scan reconstruction, a CT scan of a teapot and MR scans of a mouse embryo and a zucchini. As future work we plan to explore other integration techniques in the moving least-squares method that might allow us to reduce the amount of smoothing needed during segmentation.

## 5 ACKNOWLEDGMENT

We would like to thank Dr. Alan Barr for his technical assistance, Dr. Jason Wood of the University of Leeds for creating several useful visualization tools, and Ms. Cici Koenig for helping us with our figures. We would also like to thank Dr. John Wood of Childrens Hospital Los Angeles for providing the zucchini dataset, Dr. J. Michael Tyszka of the Division of Diagnostic Radiology at the City of Hope National Medical Center for providing the teapot dataset, Dr. Russell Jacobs of the Caltech Biological Imaging Center for providing the mouse embryo dataset, and the Caltech Multi-Res Modeling Group and the Stanford Computer Graphics Laboratory for providing the griffin dataset. This work was supported by National Science Foundation grants #ASC-89-20219, #ACI-9982273 and #ACI-0089915, and the National Institute on Drug Abuse and the National Institute of Mental Health, as part of the Human Brain Project.

## REFERENCES

- [1] D. Adalsteinsson and J.A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, pages 269–277, 1995.
- [2] M. Brejl and M. Sonka. Directional 3D edge detection in anisotropic data: Detector design and performance assessment. *Computer Vision and Image Understanding*, 77:84–110, 2000.
- [3] J.F. Canny. A computational approach to edge detection. *IEEE Trans. on Pat. Anal. and Mach. Intel.*, 8(6):679–698, 1986.
- [4] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Proc. SIGGRAPH '96*, pages 303–312, 1996.
- [5] R. Farwig. Multivariate interpolation of arbitrarily spaced data by moving least-squares methods. *J. of Comp. and App. Math.*, 16:79–93, 1986.

[6] A. Goshtasby and D. A. Turner. Fusion of short-axis and long-axis cardiac MR images. In *IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pages 202–211, San Francisco, June 1996.

[7] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*. Addison-Wesley, 1991.

[8] P. Lancaster and K. Salkauskas. Surfaces generated by moving least squares methods. *Math. Comp.*, 37:141–159, 1981.

[9] M. Leventon, O. Faugeras, W. Grimson, and W. Wells III. Level set based segmentation with intensity and curvature priors. In *Workshop on Mathematical Methods in Biomedical Image Analysis Proceedings*, pages 4–11, June 2000.

[10] R. Malladi, J.A. Sethian, and B.C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2):158–175, 1995.

[11] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London*, B(207):187–217, 1980.

[12] S. Osher and R. Fedkiw. Level set methods: An overview and some recent results. *Journal of Computational Physics*, 169:475–502, 2001.

[13] S. Osher and J. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.

[14] D. Peng, B. Merriman, S. Osher, H.-K. Zhao, and M. Kang. A PDE-based fast local level set method. *Journal of Computational Physics*, 155:410–438, 1999.

[15] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C (2nd edition)*. Cambridge University Press, New York, NY, 1992.

[16] R. Ramamoorthi and J. Arvo. Creating generative models from range images. In *SIGGRAPH '99 Proceedings*, pages 195–204, August 1999.

[17] G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, Cambridge, UK, 2001.

[18] J.A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge, UK, second edition, 1999.

[19] D. Shepard. A two-dimensional interpolation function for irregularly spaced points. In *Proc. ACM Nat. Conf.*, pages 517–524, 1968.

[20] L. Staib, X. Zeng, R. Schultz, and J. Duncan. Shape constraints in deformable models. In I. Bankman, editor, *Handbook of Medical Imaging*, chapter 9, pages 147–157. Academic Press, 2000.

[21] J.G. Tamez-Pena, S. Totterman, and K.J. Parker. MRI isotropic resolution reconstruction from two orthogonal scans. In *Proceedings SPIE Medical Imaging*, volume 4322, pages 87–97, 2001.

[22] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Proc. of SIGGRAPH '94*, pages 311–318. ACM SIGGRAPH, August 1994.

[23] R. Whitaker, D. Breen, K. Museth, and N. Soni. Segmentation of biological datasets using a level-set framework. In M. Chen and A. Kaufman, editors, *Volume Graphics 2001*, pages 249–263. Springer, Vienna, 2001.

[24] Ross T. Whitaker. Volumetric deformable models: Active blobs. In Richard A. Robb, editor, *Visualization In Biomedical Computing*, pages 122–134, Mayo Clinic, Rochester, Minnesota, 1994. SPIE.

[25] R.T. Whitaker. A level-set approach to 3D reconstruction from range data. *Int. J. of Comp. Vision*, 29(3):203–231, 1998.

[26] H.-K. Zhao, S. Osher, and R. Fedkiw. Fast surface reconstruction using the level set method. In *Proc. 1st IEEE Workshop on Variational and Level Set Methods*, pages 194–202, 2001.

## A MOVING LEAST-SQUARES

To solve for the expansion coefficients  $C$  in Eq. (10) we define the moving least-squares functional

$$E(\mathbf{x}_0) = \sum_{d=1}^D \sum_{\mathbf{x}_d} w_d(\mathbf{x}_d - \mathbf{x}_0) \left[ V_N^{(d)}(\mathbf{x}_d - \mathbf{x}_0) - V_d(\mathbf{x}_d) \right]^2 \quad (15)$$

where  $\mathbf{x}_0$  is the expansion point from where we are seeking edge information,  $V_d(\mathbf{x}_d) \equiv \widehat{V}_d(\widehat{\mathbf{x}}_d)$  and where

$$w(\mathbf{x}) \equiv \begin{cases} 1 - 2(\|\mathbf{x}\|/\Delta)^2 & \text{for } 0 \leq \|\mathbf{x}\| \leq \Delta/2 \\ 2(\|\mathbf{x}\|/\Delta - 1)^2 & \text{for } \Delta/2 < \|\mathbf{x}\| < \Delta \\ 0 & \text{for } \|\mathbf{x}\| \geq \Delta \end{cases} \quad (16)$$

is a “moving filter” that weights the contribution of different sampling points,  $\mathbf{x}_d$ , according to their Euclidean distance,  $\|\mathbf{x}_d - \mathbf{x}_0\|$ , to the expansion point,  $\mathbf{x}_0$ . Other expressions for this weighting function could of course be used, but Eq. (16) is fast to compute, has finite support (by the window parameter  $\Delta$ ), and its tangent is zero at the endpoints. After substitution of Eq. (10) into Eq. (15) we obtain the functional

$$E(\mathbf{x}_0) = \sum_{d=1}^D \sum_{\mathbf{x}_d} w_d(\mathbf{x}_d - \mathbf{x}_0) \left[ C_{000}^{(d)} - \widehat{V}_d(\mathbf{x}_d) + \sum_{i+j+k=1}^N C_{ijk}^{(0)}(x_d - x_0)^i (y_d - y_0)^j (z_d - z_0)^k \right]^2. \quad (17)$$

The minimization of this moving least-squares functional with respect to the expansion coefficients  $C$  requires the partial derivatives to vanish, *i.e.*

$$\frac{\partial \widehat{E}(\mathbf{x}_0)}{\partial C_{000}^{(d)}} = 0 = 2 \sum_{\mathbf{x}_d} w_d(\mathbf{x}_d - \mathbf{x}_0) \left[ C_{000}^{(d)} - \widehat{V}_d(\mathbf{x}_d) + \sum_{i+j+k=1}^N C_{ijk}^{(0)}(x_d - x_0)^i (y_d - y_0)^j (z_d - z_0)^k \right] \quad (18a)$$

$$\frac{\partial \widehat{E}(\mathbf{x}_0)}{\partial C_{lmn}^{(0)}} = 0 = 2 \sum_{d=1}^D \sum_{\mathbf{x}_d} w_d(\mathbf{x}_d - \mathbf{x}_0) \left[ C_{000}^{(d)} - \widehat{V}_d(\mathbf{x}_d) + \sum_{i+j+k=1}^N C_{ijk}^{(0)}(x_d - x_0)^i (y_d - y_0)^j (z_d - z_0)^k \right] \times (x_d - x_0)^l (y_d - y_0)^m (z_d - z_0)^n. \quad (18b)$$

This defines a system of linear equations in the expansion coefficients  $C_{ijk}^{(r)}$ , that can be solved using standard techniques from numerical analysis, see Eq. (11) and Eq. (13).